



# 廣東工業大學

## 课 程 设 计

课程名称 计算机网络

题目名称 Ping 程序的设计和实现

学生学院 计算机学院

专业班级 人工智能

学 号 3121005358

学生姓名 欧炜标

指导教师 姜文超

2023 年 6 月 12 日



## 计算机网络课程设计任务书

设计题目	<b>Ping 程序设计与实现(Java) ★</b>	
已知技术参数和设计要求	<p>1.已知参数：目的节点 IP 地址或主机名</p> <p>2.设计要求：通过原始套接字编程，模拟 Ping 命令，实现其基本功能，即输入一个 IP 地址或一段 IP 地址的范围，分别测试其中每个 IP 地址所对应主机的可达性，并返回耗时、生存时间等参数，并统计成功发送和回送的 Ping 报文。</p> <p>2.1 初始化 Windows Sockets 网络环境；</p> <p>2.2 解析命令行参数，构造目的端 socket 地址；</p> <p>2.3 定义 IP、ICMP 报文；</p> <p>2.4 接收 ICMP 差错报文并进行解析。</p> <p>3. 程序实现主要用到 Java 网络包中的类 InetAddress。</p>	
设计内容与步骤	<p>1.熟悉原始 socket 相关知识；</p> <p>2.熟悉 Ping 的实现原理和 ICMP 协议及报文格式；</p> <p>3.Windows Sockets 网络环境的初始化，解析命令行；</p> <p>4.创建原始套接字；</p> <p>5.构造 IP、ICMP 数据报；</p> <p>6.接收 ICMP 差错报文并进行解析处理；</p> <p>7.课程设计任务说明书。</p>	
设计工作计划与进度安排	<p>1.知识准备，整体思路设计</p> <p>2.开发环境与程序调试环境</p> <p>3. Windows Sockets 网络环境的初始化，解析命令行，创建原始套接字；</p> <p>4.构造 IP、ICMP 数据报；接收 ICMP 差错报文并进行解析处理；</p> <p>5.课程设计报告</p>	<p>2 小时</p> <p>4 小时</p> <p>8 小时</p> <p>8 小时</p> <p>4 小时</p>

## 摘要

本报告介绍了一个基于 Java 实现的 ping 程序的设计思想和关键程序说明。该程序用于探测主机之间是否可通信，通过发送 ICMP 回送请求消息给目的主机并等待其回应来判断主机是否可达。报告中解释了 ICMP 协议的工作原理以及其通过 IP 协议发送的方式。设计环境包括操作系统为 Windows 10，编程环境为 Eclipse，编程语言为 Java。

关键程序的说明部分展示了程序中的关键代码。其中，通过调用系统的 ping 命令来执行 ping 操作，并使用正则表达式匹配结果中的耗时和生存时间信息。程序通过创建 Runtime 对象和执行命令来实现与操作系统的交互。

报告还提供了程序的结构流程图和实现界面的屏幕截图。界面包括一个输入框用于输入 IP 地址或主机名，一个 Ping 按钮用于触发 ping 操作，以及一个文本域用于显示 ping 的结果。

在调试过程中，报告提到了一些问题和解决方法。例如，界面显示不完整的问题通过更新 JDK 版本解决，使用 Java 网络包中的 InetAddress 类实现 ping 功能时遇到的问题通过调用控制台的命令来解决。

测试与分析部分展示了对程序进行的各种测试，包括未输入 IP 地址时的提示、输入错误 IP 时的提示、网络连接不到目标主机的结果以及输入 IP 地址段的结果。

最后，报告总结了该课程设计的优点和不足之处，并强调了实践对于学生的重要性，通过实践能够发现问题、提高解决问题的能力，并为未来的生活和工作打下基础。

附录部分包括了程序界面的截图和代码片段。

总体而言，该报告提供了对 Java 实现的 ping 程序设计思想和关键代码的详细说明，以及程序的测试与分析结果，展示了课程设计的过程和成果。

# 目录

1. 设计思想说明.....	6
1.1 设计环境.....	6
1.2 设计思想.....	6
1.3 结构流程图.....	7
2. 关键程序说明.....	7
3. 实现界面和功能以及屏幕截图。.....	8
4. 调试过程中遇到的问题和解决的方法.....	8
5. 测试与分析.....	9
6. 用户使用说明.....	11
6.1 打开程序.....	11
6.2 输入数据.....	12
6.3 运行程序.....	12
6.4 退出程序.....	12
7. 总结.....	12
附录.....	13

# 1. 设计思想说明

## 1.1 设计环境

操作系统: win10

编程环境: Eclipse

编程语言: Java

## 1.2 设计思想

ping 程序是用来探测主机到主机之间是否可通信, 如果不能 ping 到某台主机, 表明不能和这台主机建立连接。ping 使用的是 ICMP 协议, 它发送 ICMP 回送请求消息给目的主机。ICMP 协议规定: 目的主机必须返回 ICMP 回送应答消息给源主机。如果源主机在一定时间内收到应答, 则认为主机可达。

ICMP 协议通过 IP 协议发送的, IP 协议是一种无连接的, 不可靠的数据包协议。因此, 保证数据送达的工作应该由其他的模块来完成。其中一个重要的模块就是 ICMP (网络控制报文) 协议。

当传送 IP 数据包发生错误, 比如主机不可达, 路由不可达等等, ICMP 协议将会把错误信息封包, 然后传送给主机。给主机一个处理错误的机会, 这也就是为什么说建立在 IP 层以上的协议是可能做到安全的原因。ICMP 数据包由 8bit 的错误类型和 8bit 的代码和 16bit 的校验和组成。而前 16bit 就组成了 ICMP 所要传递的信息。

PING 利用 ICMP 协议包来探测另一个主机是否可达。其原理是用类型码为 0 的 ICMP 发请求, 受到请求的主机则用类型码为 8 的 ICMP 回应。ping 程序来计算间隔时间, 并计算有多少个包被送达。用户就可以判断网络大致的情况。

本课程设计使用 java 来实现 ping 功能。 为了使用 java 来实现 ping 的功能, 可以有三种方法: 第一种是用 Java 1.5, java.net 包中的 InetAddress 实现 ICMP ping 的功能, 虽然代码简单, 但不可靠, 在 Linux 系统下会不可靠; 第二种是使用 java 调用 cmd 命令, 这种方式最简单, 可以把 ping 的过程显示在本地; 第三种也是使用 java 调用控制台的 ping 命令, 具体的思路是这样的: 通过程序调用类似 “ping 127.0.0.1 -n 10 -w 4” 的命令, 这命令会执行 ping 十次, 如果通顺则会输出类似 “来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64” 的文本 (具体数字根据实际情况会有变化), 其中中文是根据环境本地化的, 有些机器上的中文部分是英文, 但不论是中英文环境, 后面的 “<1ms TTL=62” 字样总是固定的, 它表明一次 ping 的结果是能通的。如果这个字样出现的次数等于 10 次即测试的次数, 则说明 127.0.0.1 是百分之百能连通的。技术上具体调用 dos 命令用 Runtime.getRuntime().exec 实现, 查看字符串是否符合格式用正则表达式实现。本次课设选用的是使用第三种方法。

### 1.3 结构流程图



## 2. 关键程序说明

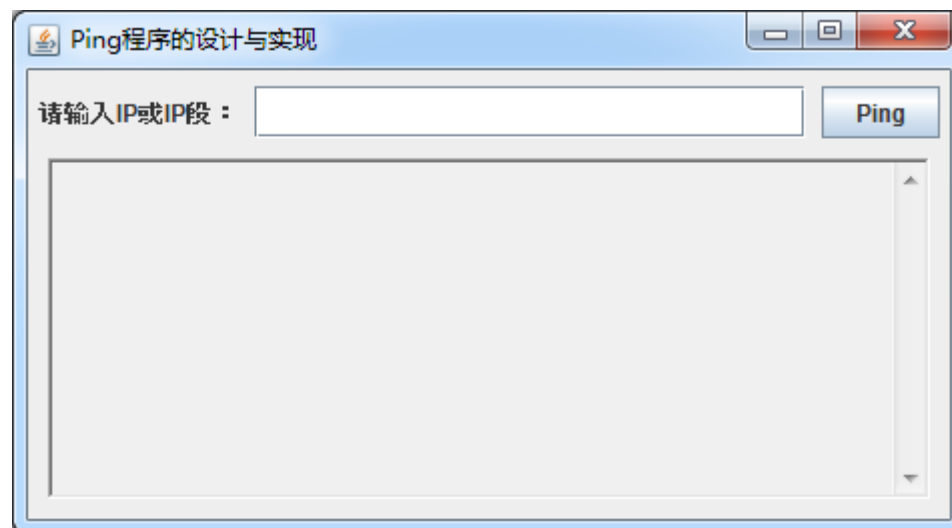
```
public Ping(String ip){
    this.ip=ip;
    runtime=Runtime.getRuntime();// 将要执行的ping命令,此命令是windows格式的命
    令
    pattern=Pattern.compile("(\\d+)ms\\s+TTL=(\\d+)",Pattern.CASE_INSENSITI
    VE); // 通过 compile 静态方法实例化 Pattern 对象 , 构造用来匹配耗时和TTL的正
    则表达式
    ttl=0;
    usage=0;
}
public boolean ping(){
    try {
```

```

//直接调用系统的ping命令，求要发送的回显请求数和等待每次回复的超时时间（ms）
Process p=runtime.exec("ping "+ip+" -n 1 -w "+TIME_OUT);
if(p==null)return false;
//从键盘接受一行输入的ip地址或IP段
BufferedReader buff=new
BufferedReader(newInputStreamReader(p.getInputStream()));
String line;
Matcher m;
while((line=buff.readLine())!=null){
m=pattern.matcher(line);
if(m.find()){
usage=Integer.parseInt(m.group(1));
ttl=Integer.parseInt(m.group(2));
return true;
}
}
} catch (IOException e) {
// TODO Auto-generated catch block
e.printStackTrace();
}
return false;
}
}

```

### 3. 实现界面和功能以及屏幕截图。



如图，在文本框中输入 IP 地址或者 IP 地址段或者主机名,点击 ping 按钮实现 ping 的过程,而下边的文本域则输入 ping 的结果,即消耗的时间和生存时间.

### 4. 调试过程中遇到的问题 and 解决的方法

问题一:界面显示一直不完整，按钮一点它按钮的字就不见了；



解决方法:更新 jdk 为较新的版本, 界面显示正常;

问题二: 刚开始用 Java 网络包中的类 `InetAddress` 实现, 不知道怎么实现 TTL, 而且即使 ping 127.0.0.1 时, 延时也达到 1s。

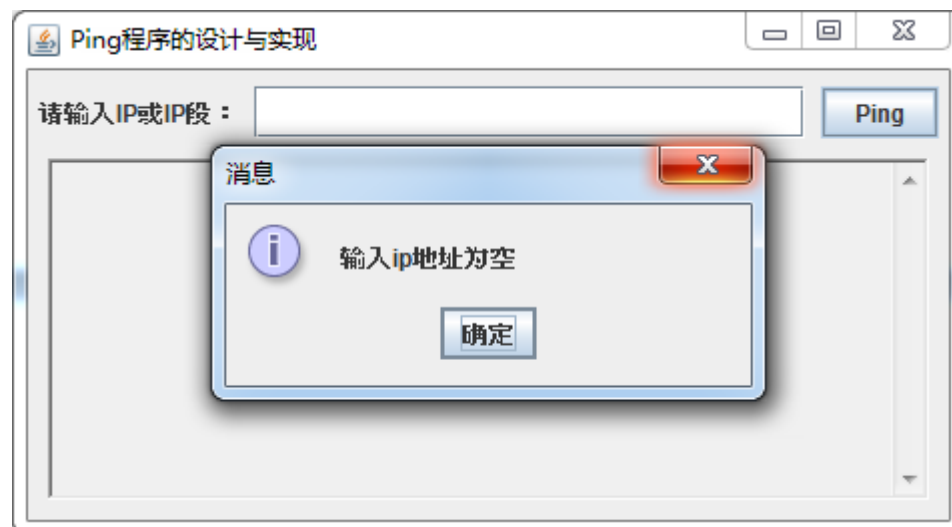
解决方法:改用调用控制台的命令, 调用 dos 命令用 `Runtime.getRuntime().exec` 实现

问题三: i p 段不知道如何一条条输出

解决方法: 把 ip 地址换成长整形的形式, 一个个入队, 再把长整形换成 I P 地址的形式一个个出队。

## 5. 测试与分析

### 1. 未输入 IP 地址时点击 Ping,提示未输入 IP



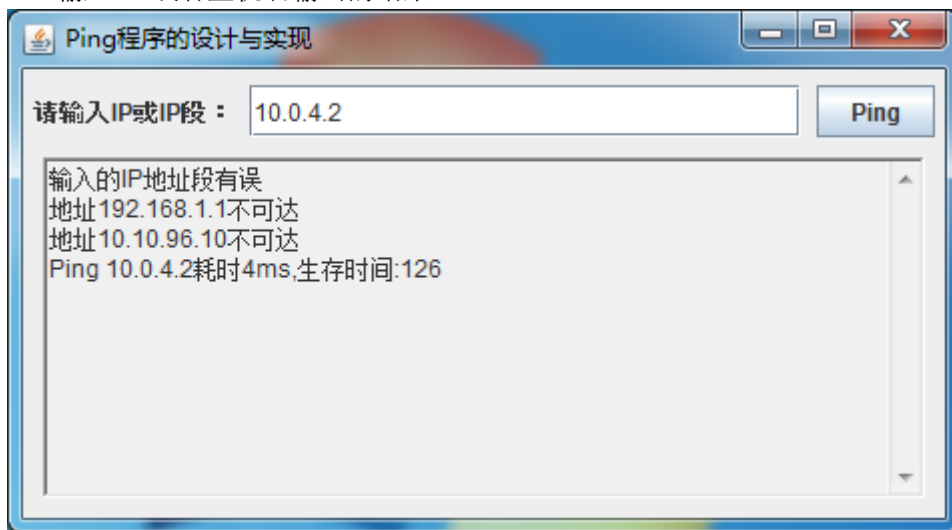
### 2. 输入错误 IP 时,显示输入 IP 有误



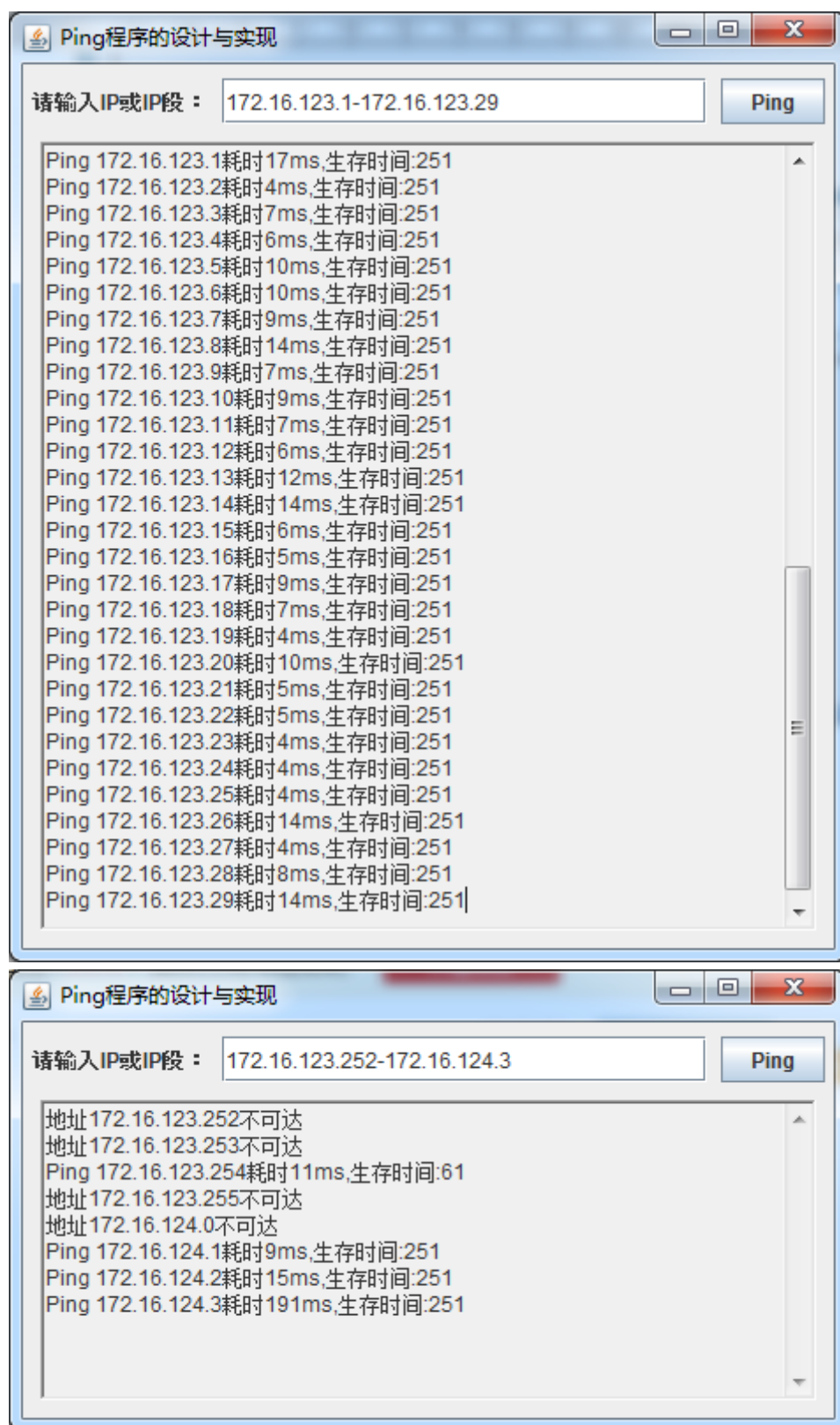
### 3. 该网络连接不到输入的 IP 时,输出结果为不可达



4. 输入 IP 或者主机名输出的结果:



5. 输入 IP 地址段输出的结果:



## 6. 用户使用说明

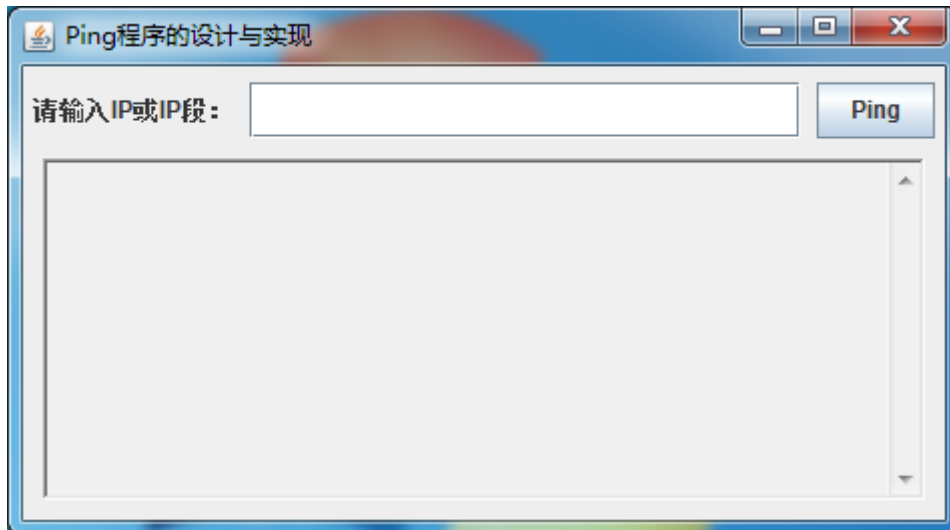
### 6.1 打开程序

点击文件夹中的 ping.jar 文件, 打开即为 ping 程序的界面;



## 6.2 输入数据

在请输入 IP 或 IP 段右边的文本框中输入您要 Ping 的 IP 地址或者 IP 地址段或者主机名；



## 6.3 运行程序

点击 Ping 按钮运行程序,会在文本域中输出相应的结果；



## 6.4 退出程序

点击程序右上角的红叉,关闭程序。

## 7. 总结

本课程设计的优点是界面简洁，功能简单易懂；不足之处是 Linux 运行用不了，因为 Linux 上的命令行 ping 出的结果与 windows 的显示不一样。

课程设计是培养学生综合运用所学知识,发现、提出、分析和解决实际问题,锻炼实践能力的重要环节,是对学生实际工作能力的具体训练和考察过程。随着科学技术发展的日新月异,当今计算机应用在生活中可以说得是无处不在,我们得多实践多动手,才能跟上知识的更新速度。此次课程设计让我收获良多。只有理论知识是远远不够的,只有把所学的理论知识与实践结合起来,从理论中得出结论,提高自己的实际动手能力和独立思考的能力。在该设计过程中我遇到了许多的问题,让我发现了自己的不足之处,对以前所学过的知识理解得不够深刻,看到了自己的实践经验还是比较缺乏,理论联系实际的能力还急需提高,只有经过不断的努力才能为以后的生活工作打下良好的基础。

## 附录

